

Nama : Mutiara Nur Balqis

NPM : 51422182

Kelas : 2IA11

Tugas : Pemrograman Berbasis Objek

Pertemuan 1-2

- **Pengenalan PBO**

Pemrograman berorientasi objek (OOP) adalah paradigma pemrograman yang menggunakan objek sebagai metode pembuatan program untuk menyelesaikan masalah yang kompleks. Objek memiliki atribut, karakter, dan kadang-kadang kondisi. OOP dikembangkan dari bahasa C dan Pascal pada tahun 1960. Pendekatan ini fokus pada objek yang dapat memecahkan masalah, bukan pada cara memecahkan masalah tersebut. OOP efektif ketika digabungkan dengan OOAD.

- **Perbedaan Antara Pemrograman Berbasis Objek dengan Pemrograman Terstruktur**

1. **Pemrograman Berbasis Objek**

- Menggabungkan fungsi dan data dalam kelas atau objek.
- Memiliki ciri Encapsulation (pengemasan), Inheritance (penurunan sifat) dan Polymorphism (perbedaan bentuk dan perilaku).
- Struktur program ringkas, cukup dengan membuat Objek dan class.
- Kode program sangat re-usable.
- Efektif digunakan untuk menyelesaikan masalah besar, karena OOP terdiri dari class-class yang memisahkan setiap kode program menjadi kelompok - kelompok kecil.
- Sulit diawal (karena harus membuat class).
- Eksekusi lebih cepat karena dieksekusi bersamaan.

2. **Pemrograman Terstruktur**

- Memecah program dalam fungsi dan data.
- Memiliki ciri Sequence (berurutan), Selection (pemilihan) dan Repetition (perulangan).
- Struktur program rumit karena berupa urutan proses dan fungsi-fungsi.
- Re-use kode program kurang.

- Tidak cocok untuk menyelesaikan masalah yang rumit, karena nantinya akan kesulitan menemukan solusi permasalahan ketika terjadi error.
- Mudah diawal, namun kompleks diproses selanjutnya.
- Eksekusi lebih lambat karena setiap perintah dikerjakan berurutan.

- Class dalam PBO

Class adalah cetak biru objek yang digunakan untuk membuat kerangka dasar. Class mirip dengan kendaraan yang memiliki ciri-ciri dan tindakan seperti merk, mesin, ban, penghidupan mesin, pemadaman mesin.

Contoh program :

```
// Deklarasi Class
public class Kendaraan{
    // Ini Adalah Class, semua konstruktor, variabel, method berada disini
}
```

- Object dalam Pemrograman

Objek dalam dunia nyata seperti orang, mobil, dan rekening bank dapat direpresentasikan dalam sebuah sistem berkas seperti rekaman. Objek memiliki nama, atribut, metode, karakteristik, dan kadang kala kondisi. Mereka dapat mempresentasikan hal-hal nyata, seperti siswa atau merek dagang, serta hal-hal konseptual, seperti font huruf. Dalam pengembangan perangkat lunak berorientasi objek, objek dalam perangkat lunak akan menyimpan state-nya dalam variable dan menyimpan informasi tingkah laku (behaviour) dalam method atau fungsi-fungsi/prosedur

Pada dasarnya ada dua karakteristik yang utama pada sebuah objek , yaitu :

1. Setiap objek memiliki atribut sebagai status yang kemudian akan disebut sebagai state.
2. Setiap objek memiliki tingkah laku yang kemudian akan disebut sebagai behaviour.

Contoh: objek sepeda. Sepeda memiliki atribut (state): pedal, roda, jeruji, dan warna. Sepeda memiliki tingkah laku (behaviour): kecepatannya menaik, kecepatannya menurun dan perpindahan gigi sepeda.

- Object Pada PBO

Untuk membuat objek, bisa menggunakan kata kunci new yang digunakan untuk membuat objek baru, selanjutnya menentukan 3 langkah untuk membuat sebuah objek. Yaitu: mendeklarasikan variable, membuat objek baru (Instansiasi) dan pemanggilan konstruktor.

Contoh program :

```
public class Kendaraan{  
    // Konstruktor Dengan Parameter  
    public Kendaraan(String nama){  
        System.out.println("Nama Kendaraannya Adalah "+ nama);  
    }  
  
    public static void main(String[] args){  
        // Perintah untuk membuat objek jenis  
        Kendaraan jenis = new Kendaraan("Pesawat Terbang");  
    }  
}  
  
// Output = Nama Kendaraannya Adalah Pesawat Terbang
```

Pertemuan 3-6

- Inheritance

1. Pengertian Dasar Inheritance

Pewarisan adalah bentuk penggunaan kembali di mana class baru dibuat dari class yang telah ada dengan penambahan fasilitas. Setiap class turunan dapat menjadi class induk untuk class turunan berikutnya. Constructor tidak diwariskan ke class turunan kecuali menggunakan perintah super. Pewarisan tunggal adalah pewarisan dari satu class pokok, sementara pewarisan ganda tidak didukung. Class yang mewariskan disebut superclass/parent class/base class, sementara class yang mewarisi disebut subclass/child class/derived class. Digunakan "extends" untuk menerapkan inheritance. Keyword "super" digunakan oleh subclass untuk memanggil constructor, atribut, dan method superclass.

2. Contoh untuk memanggil constructor milik superclass-nya :

```
super()  
super(parameter)
```

3. Contoh untuk memanggil atribut dan method milik superclass-nya :

```
super.namaAtribut  
super.namaMethod(parameter)
```

- Encapsulation

1. Pengertian Dasar Encapsulation

Encapsulation merupakan cara untuk melindungi property / method dari sebuah kelas agar tidak sembarangan diakses dan dimodifikasi oleh suatu bagian program.

2. Accessors

Cara untuk melindungi data yaitu dengan menggunakan access modifiers (hak akses). Ada 4 hak akses yang tersedia, yaitu default, public, protected, private.

No	Modifier	Pada class dan interface	Pada method dan variabel
1	Default (tidak ada modifier)	Dapat diakses oleh yang sepaket	Diwarisi oleh subkelas dipaket yang sama, dapat diakses oleh method-method yang sepaket
2	Public	Dapat diakses dimanapun	Diwarisi oleh subkelasnya, dapat diakses dimanapun
3	Protected	Tidak bisa diterapkan	Diwarisi oleh subkelasnya, dapat diakses oleh method-method yang sepaket
4	private	Tidak bisa diterapkan	Tidak dapat diakses dimanapun kecuali oleh method-method yang ada dalam kelas itu sendiri

Aksesabilitas	public	private	protected	default
Dari kelas yang sama	Ya	Ya	Ya	Ya
Dari sembarang kelas dalam paket yang sama	Ya	Tidak	Ya	Ya
Dari sembarang kelas di luar paket	Ya	Tidak	Tidak	Tidak
Dari subkelas dalam paket yang sama	Ya	Tidak	Ya	Ya
Dari subkelas di luar paket	Ya	Tidak	Ya	Tidak

- Polymorphism

1. Overloading

Overloading adalah diperbolehkannya dalam sebuah class memiliki lebih dari satu nama function/method yang sama tetapi memiliki parameter/argument yang berbeda.

```

1 public class Overloading {
2     public void Tampil(){
3         System.out.println("I love Java");
4     }
5     public void Tampil(int i){
6         System.out.println("Method dengan 1 parameter = "+i);
7     }
8     public void Tampil(int i, int j){
9         System.out.println("Method dengan 2 parameter = "+i+" & "+j);
10    }
11    public void Tampil(String str){
12        System.out.println(str);
13    }
14
15    public static void main(String a[]){
16        Overloading objek = new Overloading();
17        objek.Tampil();
18        objek.Tampil(8);
19        objek.Tampil(6,7);
20        objek.Tampil("Hello world");
21    }
22 }

```

2. Overriding

Overriding method adalah kemampuan dari subclass untuk memodifikasi method dari superclass-nya, yaitu dengan cara menumpuk (mendefinisikan kembali) method superclass-nya. Contoh overriding method dapat dilihat pada subclass “Mobil” yang mendefinisikan kembali method keterangan() dan hapus() dari class “Kendaraan”.